Improving the Security of Quantum Protocols via Commit-and-Open

Ivan Damgård[1], Serge Fehr[2], Carolin Lunemann[1], Louis Salvail[3], and Christian Schaffner[2]

[1] DAIMI, Aarhus University, Denmark
{ivan|carolin}@cs.au.dk
[2] Centrum Wiskunde & Informatica (CWI) Amsterdam, The Netherlands
{s.fehr|c.schaffner}@cwi.nl
[3] Université de Montréal (DIRO), QC, Canada
salvail@iro.umontreal.ca

Abstract. We consider two-party quantum protocols starting with a transmission of some random BB84 qubits followed by classical messages. We show a general "compiler" improving the security of such protocols: if the original protocol is secure against an "almost honest" adversary, then the compiled protocol is secure against an arbitrary computationally bounded (quantum) adversary. The compilation preserves the number of qubits sent and the number of rounds up to a constant factor. The compiler also preserves security in the bounded-quantum-storage model (BQSM), so if the original protocol was BQSM-secure, the compiled protocol can only be broken by an adversary who has large quantum memory *and* large computing power. This is in contrast to known BQSM-secure protocols, where security breaks down completely if the adversary has larger quantum memory than expected. We show how our technique can be applied to quantum identification and oblivious transfer protocols.

1 Introduction

We consider two-party quantum protocols for mutually distrusting players Alice and Bob. Such protocols typically start by Alice sending n random BB84 qubits to Bob who is supposed to measure them. Then some classical exchange of messages follows. Several protocols following this pattern have been proposed, implementing Oblivious Transfer (OT), Commitment, and Password-Based Identification [BBCS91,DFSS08,DFR$^+$07,DFSS07].

In more details, the first step of the protocol consists of Alice choosing random binary strings $x = x_1, ..., x_n$ and $\theta = \theta_1, ..., \theta_n$. She then prepares n particles where x_i is encoded in the state of the i'th particle using basis θ_i. Bob chooses a basis string $\hat{\theta} = \hat{\theta}_1, .., \hat{\theta}_n$ and measures the i'th particle in basis $\hat{\theta}_i$. If Bob plays honestly, he learns x_i whenever $\hat{\theta}_i = \theta_i$ and else gets a random independent result.

Protocols of the form we consider here are typically unconditionally secure against cheating by Alice, but can (in their basic form) be broken easily by Bob, if he does not measure the qubits immediately. This is because the protocol typically asks Alice to reveal θ at a later stage, and Bob can then measure the qubits with $\hat{\theta} = \theta$ and learn more information than he was supposed to.

In this paper, we show a general "compiler" that can be used to improve security against such an attack. We assume that the original protocol implements some two-party functionality $\mathcal{F}$ with statistical security against Bob if he is *benign*, meaning that he treats the qubits "almost honestly", a notion we make more precise below. Then we show that the compiled protocol also implements $\mathcal{F}$, but now with security against *any* computationally bounded (quantum) Bob (note that we cannot in general obtain unconditional security against both Alice and Bob, not even using quantum communication [Lo97]). The compiled protocol preserves unconditional security against Alice and has the same number of transmitted qubits and rounds as the original one up to a constant factor.

By benign behavior of Bob, we mean that after having received the qubits, two conditions are satisfied: First, Bob's quantum storage is essentially of size zero (note that it would be

exactly zero if he had measured the qubits). Second, there exists a basis string $\hat{\theta}$ such that the uncertainty about x is essentially as it would be if Bob had really measured in bases $\hat{\theta}$, namely 1 bit for every position where $\hat{\theta}$ differs from θ.

Thus, with our compiler, one can build a protocol for any two-party functionality by designing a protocol that only has to be secure if Bob is benign. We note that proofs for known protocols typically go through under this assumption. For instance, our compiler can easily be applied to the quantum identification protocols of [DFSS07] and the OT protocol of [BBCS91].

The compiler is based on a computational assumption; namely we assume the existence of a classical commitment scheme with some special properties, similar to the commitment schemes used in [DFS04] but with an additional extraction property, secure against a quantum adversary. A good candidate is the cryptosystem of Regev [Reg05]. For efficiency, we use a common reference string which allows us to use Regev's scheme in a simple way and, since it is relatively efficient, we get a protocol that is potentially practical. It is possible to generate the reference string from scratch, but this requires a more complicated non-constant round protocol [DL09].

The reader may ask whether it is really interesting to improve the security of quantum protocols for classical tasks such as identification or OT using a computational assumption. Perhaps it would be a more practical approach to use the same assumption to build *classical* protocols for the same tasks, secure against quantum attacks? To answer this, it is important to point out that our compiler also preserves security in the bounded-quantum-storage model (BQSM) [DFSS05], and this feature allows us to get security properties that classical protocols cannot achieve. In the BQSM, one assumes that Bob can only keep in his quantum memory a limited number of qubits received from Alice. With current state of the art, it is much easier to transmit and measure qubits than it is to store them for a non-negligible time, suggesting that the BQSM and the subsequently proposed noisy-quantum-storage model [WST08] are reasonable. On the other hand, if the assumption fails and the adversary can perfectly store all qubits sent, the known protocols can be easily broken. In contrast, by applying our compiler, one obtains new protocols where the adversary must have large quantum storage *and* large computing power to break the protocol.[4] [5]

The basic technique we use to construct the compiler was already suggested in connection with the first quantum OT protocol from [BBCS91]: we try to force Bob to measure by asking him to commit (using a classical scheme) to all his basis choices and measurement results, and open some of them later. While classical intuition suggests that the commitments should force Bob to measure (almost) all the qubits, it has proved very tricky to show that the approach really works against a quantum adversary. In fact, it was previously very unclear what exactly the commit-and-open approach forces Bob to do. Although some partial results for OT have been shown [Yao95,CDMS04], the original OT protocol from [BBCS91] has never been proved secure for a concrete unconditionally hiding commitment scheme – which is needed to maintain unconditional security against Alice. In this paper, we develop new quantum information-theoretic tools (that may be of independent interest) to characterize what commit-and-open achieves in general, namely it forces Bob to be benign. This property allows us to apply the compiler to any two-party functionality and in particular to show that the OT from [BBCS91] is indeed secure when using an appropriate commitment scheme.

[4] For the case of identification[DFSS07], the compiled protocol is not only secure against adversaries trying to impersonate Alice or Bob, but can also be made secure against man-in-the-middle attacks, where again the adversary must have large quantum storage and large computing power to break the protocol.

[5] One may try to achieve the same security by combining one of the previous BQSM secure protocols with a computationally secure classical protocol, but it is not clear that this technique will work for all functionalities, and it would require independent key material for the two instances. For the case of password-based identification it would require users to have two passwords.

2 Preliminaries

We assume the reader to be familiar with the basic notation and concepts of quantum information processing [NC00]. In this paper, the computational or $+$-basis is defined by the pair $\{|0\rangle, |1\rangle\}$ (also written as $\{|0\rangle_+, |1\rangle_+\}$). The pair $\{|0\rangle_\times, |1\rangle_\times\}$ denotes the diagonal or $\times$-basis, where $|0\rangle_\times = (|0\rangle + |1\rangle)/\sqrt{2}$ and $|1\rangle_\times = (|0\rangle - |1\rangle)/\sqrt{2}$. We write $|x\rangle_\theta = |x_1\rangle_{\theta_1} \otimes \cdots \otimes |x_n\rangle_{\theta_n}$ for the n-qubit state where string $x = (x_1, \ldots, x_n) \in \{0,1\}^n$ is encoded in bases $\theta = (\theta_1, \ldots, \theta_n) \in \{+, \times\}^n$. For $S \subseteq \{1, \ldots, n\}$ of size s, we denote by $\bar{S} := \{1, \ldots, n\} \backslash S$ the complement of S and define $x|_S \in \{0,1\}^s$ and $\theta|_S \in \{+, \times\}^s$ to be the restrictions $(x_i)_{i \in S}$ and $(\theta_i)_{i \in S}$, respectively. For two strings $x, y \in \{0,1\}^n$, we define the *Hamming distance* between x and y as $d_H(x,y) := |\{i : x_i \neq y_i\}|$.

We use upper case letters for the random variables in the proofs that describe the respective values in the protocol. Given a bipartite quantum state ρ_{XE}, we say that X is *classical* if ρ_{XE} is of the form $\rho_{XE} = \sum_{x \in \mathcal{X}} P_X(x) |x\rangle\langle x| \otimes \rho_E^x$ for a probability distribution P_X over a finite set $\mathcal{X}$, i.e. the state of the quantum register E depends on the classical random variable X in the sense that E is in state ρ_E^x exactly if $X = x$. This naturally extends to states with two or more classical registers.

For a state ρ_{XE} as above, X is *independent* of register E if $\rho_{XE} = \rho_X \otimes \rho_E$, where $\rho_X = \sum_x P_X(x) |x\rangle\langle x|$ and $\rho_E = \sum_x P_X(x) \rho_E^x$. We also need to express that a random variable X is independent of a quantum state E *when given a random variable* Y. Independence means that when given Y, the state E gives no additional information on X. Formally, adopting the notion introduced in [DFSS07], we require that ρ_{XYE} equals $\rho_{X \leftrightarrow Y \leftrightarrow E}$, where the latter is defined as

$$\rho_{X \leftrightarrow Y \leftrightarrow E} := \sum_{x,y} P_{XY}(x,y) |x\rangle\langle x| \otimes |y\rangle\langle y| \otimes \rho_E^y \,,$$

where $\rho_E^y := \sum_x P_{X|Y}(x|y) \rho_E^{x,y}$. In other words, $\rho_{XYE} = \rho_{X \leftrightarrow Y \leftrightarrow E}$ precisely if $\rho_E^{x,y} = \rho_E^y$ for all x and y.

Full (conditional) independence is often too strong a requirement, and it usually suffices to be "close" to such a situation. Closeness of two states ρ and σ is measured in terms of their trace distance $\delta(\rho, \sigma) = \frac{1}{2} \operatorname{tr}(|\rho - \sigma|)$, where for any operator A, $|A|$ is defined as $|A| := \sqrt{AA^\dagger}$.

A quantum algorithm consists of a family $\{C_n\}_{n \in \mathbb{N}}$ of quantum circuits and is said to run in *polynomial time*, if the number of gates of C_n is polynomial in n. Two families of quantum states $\{\rho_n\}_{n \in \mathbb{N}}$ and $\{\sigma_n\}_{n \in \mathbb{N}}$ are called *quantum-computationally indistinguishable*, denoted $\rho \overset{q}{\approx} \sigma$, if any polynomial-time quantum algorithm has negligible advantage (in n) of distinguishing ρ_n from σ_n. Analogously, we call them *statistically indistinguishable*, $\rho \overset{s}{\approx} \sigma$, if their trace distance $\delta(\rho_n, \sigma_n)$ is negligible in n.

Definition 2.1 (Min-Entropy). *The* min-entropy *of a random variable X with probability distribution P_X is defined as $H_\infty(X) := -\log\big(\max_x P_X(x)\big)$.*

Definition 2.2 (Max-Entropy). *The* max-entropy *of a density matrix ρ is defined as $H_0(\rho) := \log\big(\operatorname{rank}(\rho)\big)$.*

We will make use of the following properties of a pure state that can be written as a "small superposition" of basis vectors.

Lemma 2.3. *Let $|\varphi_{AE}\rangle \in \mathcal{H}_A \otimes \mathcal{H}_E$ be of the form $|\varphi_{AE}\rangle = \sum_{i \in J} \alpha_i |i\rangle |\varphi_E^i\rangle$, where $\{|i\rangle\}_{i \in I}$ is a basis of $\mathcal{H}_A$ and $J \subseteq I$. Then, the following holds.*

1. *Let $\tilde{\rho}_{AE} = \sum_{i \in J} |\alpha_i|^2 |i\rangle\langle i| \otimes |\varphi_E^i\rangle\langle\varphi_E^i|$, and let W and $\tilde{W}$ be the outcome of measuring A of $|\varphi_{AE}\rangle$ respectively of $\tilde{\rho}_{AE}$ in some basis $\{|w\rangle\}_{w \in \mathcal{W}}$. Then,*[6]

$$H_\infty(W) \geq H_\infty(\tilde{W}) - \log |J| \,.$$

[6] Using Renner's definition for conditional min-entropy [Ren05], one can actually show that $H_\infty(W|E) \geq H_\infty(\tilde{W}|E) - \log |J|$.

2. The reduced density matrix $\rho_E = \mathrm{tr}_A(|\varphi_{AE}\rangle\langle\varphi_{AE}|)$ has max-entropy

$$H_0(\rho_E) \leq \log |J| \,.$$

Proof. For 1., we may understand $\tilde{\rho}_{AE}$ as being in state $|i\rangle|\varphi_E^i\rangle$ with probability $|\alpha_i|^2$, so that we easily see that

$$\begin{aligned} P_{\tilde{W}}(w) &= \sum_{i\in J} |\alpha_i|^2 |\langle w|i\rangle|^2 = \sum_{i\in J} |\alpha_i|^2 |\langle w|i\rangle|^2 \cdot \sum_{i\in J} 1^2 \cdot \frac{1}{|J|} \\ &\geq \left| \sum_{i\in J} \alpha_i \langle w|i\rangle \right|^2 \cdot \frac{1}{|J|} = \left| \langle w| \sum_{i\in J} \alpha_i |i\rangle \right|^2 \cdot \frac{1}{|J|} = P_W(w) \cdot \frac{1}{|J|} \,, \end{aligned}$$

where the inequality is Cauchy-Schwartz. This proves 1.

For 2., note that $\rho_E = \mathrm{tr}_A(|\varphi_{AE}\rangle\langle\varphi_{AE}|) = \sum_{i\in J} |\alpha_i|^2 |\varphi_E^i\rangle\langle\varphi_E^i|$. The claim follows immediately from the sub-additivity of the rank:

$$\mathrm{rank}(\rho_E) \leq \sum_{i\in J} \mathrm{rank}(|\alpha_i|^2 |\varphi_E^i\rangle\langle\varphi_E^i|) \leq \sum_{i\in J} 1 = |J| \,,$$

where we use that the $|\varphi_E^i\rangle\langle\varphi_E^i|$'s have rank at most 1. □

3 Definition of Security

In order to define security of our two-party protocols, we follow the framework put forward by Fehr and Schaffner in [FS09]. We are interested in quantum protocols that implement *classical functionalities* such as oblivious transfer. Such primitives are often used as building blocks in more complicated classical (multi-party) protocols which implement advanced tasks. Therefore, it is natural to restrict our focus on quantum protocols that run in a classical environment and have classical in- and outputs. A two-party quantum protocol $\Pi = (\mathsf{A}_m, \mathsf{B}_m)$ consists of an infinite family of interactive quantum circuits for players Alice and Bob indexed by the security parameter m (in our case, m will also be the number of qubits transmitted). To ease notation, we often leave the dependence on m implicit. A classical non-reactive two-party *ideal functionality* $\mathcal{F}$ is given by a conditional probability distribution $P_{\mathcal{F}(U,V)|UV}$, inducing a pair of random variables $(X, Y) = \mathcal{F}(U, V)$ for every joint distribution of U and V. The definition of correctness of a protocol is straightforward.

Definition 3.1 (Correctness). *A protocol $\Pi = (\mathsf{A}, \mathsf{B})$* correctly implements *an ideal classical functionality $\mathcal{F}$, if for every distribution of the input values U and V, the resulting common output satisfies*

$$(U, V, (X, Y)) \overset{s}{\approx} (U, V, \mathcal{F}(U, V)) \,.$$

Let us denote by $out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}}$ the joint output[7] of the "ideal-life" protocol, where Alice and Bob forward their inputs to $\mathcal{F}$ and output whatever they obtain from $\mathcal{F}$. And we write $out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'}$ for the joint output of the execution of this protocol with a dishonest Bob with strategy $\hat{\mathsf{B}}'$ (and similarly for a dishonest Alice). Note that Bob's possibilities in the ideal world are very limited: he can produce some classical input V for $\mathcal{F}$ from his input quantum state V', and then he can prepare and output a quantum state Y' which might depend on $\mathcal{F}$'s classical reply Y.

[7] We use a slightly different notation here than in [FS09]. Our notation $out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}}$ does not mention the name of the input registers and corresponds to $(\mathcal{F}_{\hat{\mathsf{A}},\hat{\mathsf{B}}})\rho_{UV}$ in [FS09].

3.1 Information-Theoretic Security

We define information-theoretic security using the real/ideal-world paradigm, which requires that by attacking a protocol in the real world the dishonest party cannot achieve (significantly) more than when attacking the corresponding functionality in the ideal world. To be consistent with the framework used in [FS09], we restrict the joint input state, consisting of a classical input to the honest party and a possibly quantum input to the dishonest party, to a special form: in case of a dishonest Bob (and correspondingly for a dishonest Alice), we require that Bob's input consists of a classical part Z and a quantum part V', such that the joint state $\rho_{UZV'}$ satisfies $\rho_{UZV'} = \rho_{U\leftrightarrow Z\leftrightarrow V'}$, i.e., that V' is correlated with Alice's input only via the classical Z. We call a joint input state of that form (respectively of the form $\rho_{U'ZV} = \rho_{U'\leftrightarrow Z\leftrightarrow V}$ in case of dishonest Alice) a *legitimate* input state. As shown in [FS09], this restriction on the input state leads to a meaningful security definition with a composition theorem that guarantees sequential composition within *classical* outer protocols. Furthermore, the results of Section 4 also hold when quantifying over all (possibly non-legitimate) joint input states.

Definition 3.2 (Unconditional security against dishonest Alice). *A protocol $\Pi = (\mathsf{A}, \mathsf{B})$ implements an ideal classical functionality $\mathcal{F}$ unconditionally securely against dishonest Alice, if for any real-world adversary A' there exists an ideal-world adversary $\hat{\mathsf{A}}'$ such that for any legitimate input state, it holds that the outputs in the real and ideal world are statistically indistinguishable, i.e.*

$$out^{\Pi}_{\mathsf{A}',\mathsf{B}} \stackrel{s}{\approx} out^{\mathcal{F}}_{\hat{\mathsf{A}}',\hat{\mathsf{B}}}\,.$$

Definition 3.3 (Unconditional security against dishonest Bob). *A protocol $\Pi = (\mathsf{A}, \mathsf{B})$ implements an ideal classical functionality $\mathcal{F}$ unconditionally securely against dishonest Bob, if for any real-world adversary B' there exists an ideal-world adversary $\hat{\mathsf{B}}'$ such that for any legitimate input state, it holds that the outputs in the real and ideal world are statistically indistinguishable, i.e.*

$$out^{\Pi}_{\mathsf{A},\mathsf{B}'} \stackrel{s}{\approx} out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'}\,.$$

It has been shown in Theorem 5.1 in [FS09] that protocols fulfilling the above definitions compose sequentially as follows. For a classical real-life protocol Σ which makes at most k oracle calls to functionalities $\mathcal{F}_1, \ldots, \mathcal{F}_k$, it is guaranteed that whatever output Σ produces, the output produced when the oracle calls are replaced by ε-secure protocols is at distance at most $O(k\varepsilon)$.

Notice that in the definitions above, we do *not* require the running time of ideal-world adversaries to be polynomial whenever the real-life adversaries run in polynomial time. This way of defining unconditional security can lead to the (unwanted) effect that unconditional security does not necessarily imply computational security. However, for the security of the construction proposed in this paper, efficient ideal-life adversaries can be guaranteed, as discussed in Section 5.3.

3.2 Computational Security in the CRS Model

One can define security against a computationally bounded dishonest Bob analogously to information-theoretic security with the two differences that the input given to the parties has to be sampled by an efficient quantum algorithm and that the output states should be computationally indistinguishable.

In the common-reference-string (CRS) model, all participants in the real-life protocol $\Pi_{\mathsf{A},\mathsf{B}}$ have access to a classical public string ω which is chosen before any interaction starts according to a distribution only depending on the security parameter. On the other hand, the participants in the "ideal-life" protocol $\mathcal{F}_{\hat{\mathsf{A}},\hat{\mathsf{B}}}$ interacting only with the ideal functionality do not make use of the string ω. Hence, an ideal-world adversary $\hat{\mathsf{B}}'$, that operates by simulating the real world to the adversary B', is free to choose ω in any way he wishes.

In order to define computational security against a dishonest Bob in the CRS model, we consider a polynomial-size quantum circuit, called *input sampler*, which takes as input the security parameter m and the CRS ω (chosen according to its distribution) and produces the input state $\rho_{UZV'}$; U is Alice's classical input to the protocol, and Z and V' denote the respective classical and quantum information given to dishonest Bob. We call the input sampler *legitimate* if $\rho_{UZV'} = \rho_{U\leftrightarrow Z\leftrightarrow V'}$.

In the following and throughout the article, we let $\mathfrak{B}_{\text{poly}}$ be the family of all *polynomial-time* quantum strategies for dishonest Bob B'.

Definition 3.4 (Computational security against dishonest Bob). *A protocol $\Pi = (\mathsf{A}, \mathsf{B})$ implements an ideal classical functionality $\mathcal{F}$ computationally securely against dishonest Bob, if for any real-world adversary $\mathsf{B}' \in \mathfrak{B}_{\text{poly}}$ who has access to the common reference string ω, there exists an ideal-world adversary $\hat{\mathsf{B}}' \in \mathfrak{B}_{\text{poly}}$ not using ω such that for any efficient legitimate input sampler, it holds that the outputs in the real and ideal world are q-indistinguishable, i.e.*

$$out^{\Pi}_{\mathsf{A},\mathsf{B}'} \overset{q}{\approx} out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'} .$$

In Appendix A, we show that also the computational security definition, as given here, allows for (sequential) composition of quantum protocols into classical outer protocols.

4 Improving the Security via Commit-and-Open

4.1 Security against Benign Bob

In this paper, we consider quantum two-party protocols that follow a particular but very typical construction design. These protocols consist of two phases, called *preparation* and *post-processing* phase, and are as specified in Figure 1. We call a protocol that follows this construction design a *BB84-type* protocol.

PROTOCOL Π

Preparation: A chooses $x \in_R \{0,1\}^n$ and $\theta \in_R \{+,\times\}^n$ and sends $|x\rangle_\theta$ to B, and B chooses $\hat{\theta} \in_R \{+,\times\}^n$ and obtains $\hat{x} \in \{0,1\}^n$ by measuring $|x\rangle_\theta$ in basis $\hat{\theta}$.

Post-processing: Arbitrary classical communication and local computations.

Fig. 1. Generic BB84-type quantum protocol Π.

The following definition captures information-theoretic security against a somewhat mildly dishonest Bob who we call a *benign* (dishonest) Bob. Such a dishonest Bob is benign in that, in the preparation phase, he does not deviate too much from what he is supposed to do; in the post-processing phase though, he may be arbitrarily dishonest.

To make this description formal, we fix an arbitrary choice of θ and an arbitrary value for the classical information, z, which B' may obtain as a result of the preparation phase (i.e. $z = (\hat{\theta}, \hat{x})$ in case B' is actually honest). Let X denote the random variable describing the bit-string x, where we understand the distribution P_X of X to be conditioned on the fixed values of θ and z. Furthermore, let ρ_E be the state of B''s quantum register E after the preparation phase. Note that, still with fixed θ and z, ρ_E is of the form $\rho_E = \sum_x P_X(x)\rho^x_E$, where ρ^x_E is the state of B''s quantum register in case X takes on the value x. In general, the ρ^x_E's may be mixed, but we can think of them as being reduced pure states: $\rho^x_E = \text{tr}_R(|\psi^x_{ER}\rangle\langle\psi^x_{ER}|)$ for a suitable register R and pure states $|\psi^x_{ER}\rangle$; we then call the state $\rho_{ER} = \sum_x P_X(x)|\psi^x_{ER}\rangle\langle\psi^x_{ER}|$ a *pointwise purification* (with respect to X) of ρ_E.

Obviously, in case B' is honest, X_i is fully random whenever $\theta_i \neq \hat{\theta}_i$, so that $H_\infty\big(X|_I \,\big|\, X|_{\bar{I}} = x|_{\bar{I}}\big) = d_H\big(\theta|_I, \hat{\theta}|_I\big)$ for every $I \subseteq \{1,\ldots,n\}$ and every $x|_I$, and B' does not store any non-trivial quantum state so that R is "empty" and $H_0(\rho_{ER}) = H_0(\rho_E) = 0$. A benign Bob B' is now specified to behave close-to-honestly in the preparation phase: he produces an auxiliary output $\hat{\theta}$ after the preparation phase, and given this output, we are in a certain sense close to the ideal situation where Bob really measured in basis $\hat{\theta}$ as far as the values of $H_\infty\big(X|_I \,\big|\, X|_{\bar{I}} = x|_{\bar{I}}\big)$ and $H_0(\rho_{ER})$ are concerned.[8] We now make this precise:

Definition 4.1 (Unconditional security against *benign* Bob). *A BB84-type quantum protocol Π securely implements $\mathcal{F}$ against a β-*benign *Bob for some parameter $\beta \geq 0$, if it securely implements $\mathcal{F}$ according to Definition 3.3, with the following two modifications:*

1. *The quantification is over all B' with the following property: after the preparation phase B' either aborts, or else produces an auxiliary output $\hat{\theta} \in \{+,\times\}^n$. Moreover, the joint state of A, B' (after $\hat{\theta}$ has been output) is statistically indistinguishable from a state for which it holds that for any fixed values of θ, $\hat{\theta}$ and z, for any subset $I \subseteq \{1,\ldots,n\}$, and for any $x|_{\bar{I}}$*

$$H_\infty\big(X|_I \,\big|\, X|_{\bar{I}} = x|_{\bar{I}}\big) \geq d_H\big(\theta|_I, \hat{\theta}|_I\big) - \beta n \qquad \text{and} \qquad H_0\big(\rho_{ER}\big) \leq \beta n \tag{1}$$

   where ρ_{ER} is the pointwise purification of ρ_E with respect to X.
2. *$\hat{\mathsf{B}}'$'s running-time is polynomial in the running-time of B'.*

4.2 From Benign to Computational Security

We show a generic compiler which transforms any BB84-type protocol into a new quantum protocol for the same task. The compiler achieves that if the original protocol is unconditionally secure against dishonest Alice and unconditionally secure against *benign* Bob, then the compiled protocol is still unconditionally secure against dishonest Alice and it is *computationally secure* against *arbitrary* dishonest Bob.

The idea behind the construction of the compiler is to incorporate a commitment scheme and force Bob to behave benignly by means of a commit-and-open procedure. Figure 2 shows the compilation of an arbitrary BB84-type protocol Π. The quantum communication is increased from n to $m = n/(1-\alpha)$ qubits, where $0 < \alpha < 1$ is some additional parameter that can be arbitrarily chosen. The compiled protocol also requires 3 more rounds of interaction.

PROTOCOL $\mathcal{C}^\alpha(\Pi)$

Preparation: A chooses $x \in_R \{0,1\}^m$ and $\theta \in_R \{+,\times\}^m$ and sends $|x\rangle_\theta$ to B. Then, B chooses $\hat{\theta} \in_R \{+,\times\}^m$ and obtains $\hat{x} \in \{0,1\}^m$ by measuring $|x\rangle_\theta$ in basis $\hat{\theta}$.

Verification:
1. B commits to $\hat{\theta}$ and $\hat{x}$ position-wise: $c_i := \mathtt{Commit}\big((\hat{\theta}_i, \hat{x}_i), r_i\big)$ with randomness r_i for $i = 1,\ldots,m$. He sends the commitments to A.
2. A sends a random test subset $T \subset \{1,\ldots,m\}$ of size αm. B opens c_i for all $i \in T$, and A checks that the openings were correct and that $x_i = \hat{x}_i$ whenever $\theta_i = \hat{\theta}_i$. If all tests are passed, A accepts, otherwise, she rejects and aborts.
3. The tested positions are discarded by both parties: A and B restrict x and θ, respectively $\hat{\theta}$ and $\hat{x}$, to $i \in \bar{T}$.

Post-processing: As in Π (with $x, \theta, \hat{x}$ and $\hat{\theta}$ restricted to the positions $i \in \bar{T}$).

Fig. 2. Compiled protocol $\mathcal{C}^\alpha(\Pi)$.

[8] The reason why we consider the *pointwise purification* of ρ_E is to prevent Bob from artificially blowing up $H_0(\rho_{ER})$ by locally generating a large mixture or storing an unrelated mixed input state.

We need to specify what kind of commitment scheme to use. In order to preserve unconditional security against dishonest Alice, the commitment scheme needs to be unconditionally hiding, and so can at best be computationally binding. However, for a plain computationally binding commitment scheme, we do not know how to reduce the computational security of $\mathcal{C}^\alpha(\Pi)$ against dishonest Bob to the computational binding property of the commitment scheme.[9] Therefore, we use a commitment scheme with additional properties: we require a *keyed* commitment scheme $\mathtt{Commit}_{\mathtt{pk}}$, where the corresponding public key $\mathtt{pk}$ is generated by one of two possible key-generation algorithms: $\mathcal{G}_{\mathtt{H}}$ or $\mathcal{G}_{\mathtt{B}}$. For a key $\mathtt{pkH}$ generated by $\mathcal{G}_{\mathtt{H}}$, the commitment scheme $\mathtt{Commit}_{\mathtt{pkH}}$ is unconditionally hiding, whereas the other generator, $\mathcal{G}_{\mathtt{B}}$, actually produces a key *pair* $(\mathtt{pkB}, \mathtt{sk})$, so that the secret key $\mathtt{sk}$ allows to efficiently extract m from $\mathtt{Commit}_{\mathtt{pkB}}(m, r)$, and as such $\mathtt{Commit}_{\mathtt{pkB}}$ is unconditionally binding. Furthermore, we require $\mathtt{pkH}$ and $\mathtt{pkB}$ to be computationally indistinguishable, even against quantum attacks. We call such a commitment scheme a *dual-mode* commitment scheme.[10] As a candidate for implementing such a system, we propose the public-key encryption scheme of Regev [Reg05], which is based on a worst-case lattice assumption and is not known to be breakable even by (efficient) quantum algorithms. Regev does not explicitly state that the scheme has the property we need, but this is implicit in his proof that the underlying computational assumption implies semantic security.[11]

For simplicity and efficiency, we consider the common-reference-string model, and we assume the key $\mathtt{pkB}$ for the commitment scheme, generated according to $\mathcal{G}_{\mathtt{B}}$, to be contained in the CRS. We sketch in Section 5.4 how to avoid the CRS model, at the cost of a non constant-round construction where the parties generate the CRS jointly by means of a coin-tossing protocol (see [DL09] for details).

We sometimes write $\mathcal{C}^\alpha_{\mathtt{pkH}}(\Pi)$ for the compiled protocol $\mathcal{C}^\alpha(\Pi)$ to stress that a key $\mathtt{pkH}$ produced by $\mathcal{G}_{\mathtt{H}}$ is used for the dual-mode commitment scheme, and we write $\mathcal{C}^\alpha_{\mathtt{pkB}}(\Pi)$ when a key $\mathtt{pkB}$ produced by $\mathcal{G}_{\mathtt{B}}$ is used instead.

Theorem 4.2. *Let Π be a BB84-type protocol, unconditionally secure against dishonest Alice and against β-benign Bob for some constant $\beta > 0$. Consider the compiled protocol $\mathcal{C}^\alpha(\Pi)$ for an arbitrary $\alpha > 0$, where the commitment scheme is instantiated by a dual-mode commitment scheme as described above. Then, $\mathcal{C}^\alpha(\Pi)$ is unconditionally secure against dishonest Alice and computationally secure against dishonest Bob in the CRS model.*

We now prove this theorem, which assumes noise-free quantum communication; we explain in Section 5.1 how to generalize it for a noisy quantum channel. Correctness is obvious. In order to show unconditional security against dishonest Alice, we notice that the unconditional hiding property of the commitment scheme ensures that dishonest Alice does not learn any additional information. Furthermore, as the ideal-life adversary $\hat{\mathsf{A}}'$ is not required to be time-bounded by Definition 3.2, she can break the binding-property of the commitment scheme and thereby perfectly simulate the behavior of an honest Bob towards $\hat{\mathsf{A}}$ attacking $\mathcal{C}^\alpha(\Pi)$. The issue of efficiency of the ideal-life adversaries is addressed in Section 5.3.

As for computational security against dishonest Bob, according to Definition 3.4, we need to prove that for every real-world adversary $\mathsf{B}' \in \mathfrak{B}_{\mathrm{poly}}$ attacking $\mathcal{C}^\alpha(\Pi)$, there exists a suitable ideal-world adversary $\hat{\mathsf{B}}' \in \mathfrak{B}_{\mathrm{poly}}$ attacking $\mathcal{F}$ such that

$$out^{\mathcal{C}^\alpha(\Pi)}_{\mathsf{A},\mathsf{B}'} \overset{q}{\approx} out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'} .$$

[9] Classically, this would be done by a rewinding argument, but this fails to work for a quantum Bob.

[10] The notions of dual-mode *cryptosystems* and of meaningful/meaningless encryptions, as introduced in [PVW08] and [KN08], are similar in spirit but differ slightly technically.

[11] The proof compares the case where the public key is generated normally to a case where it is chosen with no relation to any secret key. It is then argued that the assumption implies that the two cases are computationally indistinguishable, and that in the second case, a ciphertext carries essentially no information about the message. This argument implies what we need.

First, note that by the computational indistinguishability of $\mathtt{pkH}$ and $\mathtt{pkB}$,

$$out^{\mathcal{C}^\alpha(\Pi)}_{\mathsf{A},\mathsf{B}'} = out^{\mathcal{C}^\alpha_{\mathtt{pkH}}(\Pi)}_{\mathsf{A},\mathsf{B}'} \overset{q}{\approx} out^{\mathcal{C}^\alpha_{\mathtt{pkB}}(\Pi)}_{\mathsf{A},\mathsf{B}'} . \tag{2}$$

Then, we construct an adversary $\mathsf{B}'_\circ \in \mathfrak{B}_{\text{poly}}$ who attacks the unconditional security against benign Bob of protocol Π, and which satisfies

$$out^{\mathcal{C}^\alpha_{\mathtt{pkB}}(\Pi)}_{\mathsf{A},\mathsf{B}'} = out^{\Pi}_{\mathsf{A}_\circ,\mathsf{B}'_\circ} , \tag{3}$$

where $\mathsf{A}_\circ$ honestly executes Π. We define $\mathsf{B}'_\circ$ in the following way. Consider the execution of $\mathcal{C}^\alpha(\Pi)$ between A and B'. We split A into two players $\mathsf{A}_\circ$ and $\tilde{\mathsf{A}}$, where we think of $\tilde{\mathsf{A}}$ as being placed in between $\mathsf{A}_\circ$ and B', see Figure 3. $\mathsf{A}_\circ$ plays honest Alice's part of Π while $\tilde{\mathsf{A}}$ acts as follows: It receives n qubits from $\mathsf{A}_\circ$, produces $\alpha n/(1-\alpha)$ random BB84 qubits of its own and interleaves them randomly with those received and sends the resulting $m = n/(1-\alpha)$ qubits to B'. It then does the verification step of $\mathcal{C}^\alpha(\Pi)$ with B', asking to have commitments corresponding to its own qubits opened. If this results in accept, it lets $\mathsf{A}_\circ$ finish the protocol with B'. Note that the pair $(\mathsf{A}_\circ, \tilde{\mathsf{A}})$ does exactly the same as A; however, we can also move the actions of $\tilde{\mathsf{A}}$ to Bob's side, and define $\mathsf{B}'_\circ$ as follows. $\mathsf{B}'_\circ$ samples $(\mathtt{pkB}, \mathtt{sk})$ according to $\mathcal{G}_\mathsf{B}$ and executes Π with A by locally running $\tilde{\mathsf{A}}$ and B', using $\mathtt{pkB}$ as CRS. If $\tilde{\mathsf{A}}$ accepts the verification then $\mathsf{B}'_\circ$ outputs $\hat{\theta} \in \{0,1\}^n$ (as required from a *benign* Bob), obtained by decrypting the unopened commitments with the help of $\mathtt{sk}$; else, $\mathsf{B}'_\circ$ aborts at this point. It is now clear that Equation (3) holds: exactly the same computation takes place in both "experiments", the only difference being that they are executed partly by different entities. The last step is to show that

$$out^{\Pi}_{\mathsf{A}_\circ,\mathsf{B}'_\circ} \overset{s}{\approx} out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'} , \tag{4}$$

for some $\hat{\mathsf{B}}'$. It is clear that the theorem follows from (2) - (4) together.

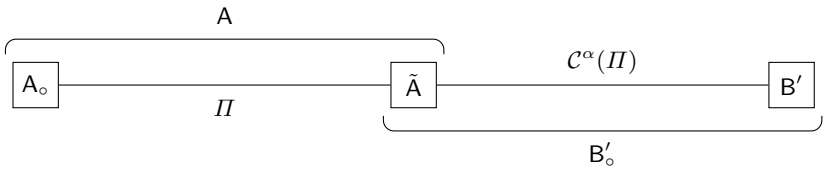

Fig. 3. Constructing an attacker $\mathsf{B}'_\circ$ against Π from an attacker B' against $\mathcal{C}^\alpha(\Pi)$.

Now (4) actually claims that $\hat{\mathsf{A}}, \hat{\mathsf{B}}'$ successfully simulate $\mathsf{A}_\circ$ and $\mathsf{B}'_\circ$ executing Π, and this claim follows by assumption of benign security of Π if we show that $\mathsf{B}'_\circ$ is β-benign according to Definition 4.1 for any $\beta > 0$. We show this in the following subsection, i.e., the joint state of $\mathsf{A}_\circ, \mathsf{B}'_\circ$ after the preparation phase is statistically indistinguishable from a state ρ_{Ideal} which satisfies the bounds (1) from Definition 4.1.

4.3 Completing the Proof: Bounding Entropy and Memory Size

First recall that $\mathsf{A}_\circ$ executing Π with $\mathsf{B}'_\circ$ can equivalently be thought of as A executing $\mathcal{C}^\alpha_{\mathtt{pkB}}(\Pi)$ with B'. Furthermore, a joint state of A, B' is clearly also a joint state of $\mathsf{A}_\circ, \mathsf{B}'_\circ$.

To show the existence of ρ_{Ideal} as promised above, it therefore suffices to show such a state for A, B'. In other words, we need to show that the execution of $\mathcal{C}^\alpha_{\mathtt{pkB}}(\Pi)$ with honest Alice A and arbitrarily dishonest Bob B' will, after verification, be close to a state where (1) holds. To show this closeness, we consider an equivalent EPR-pair version, where Alice creates m EPR pairs $(|00\rangle + |11\rangle)/\sqrt{2}$, sends one qubit in each pair to Bob and keeps the others in register

A. Alice measures her qubits only when needed: she measures the qubits within T in Step 2 of the verification phase, and the remaining qubits at the end of the verification phase. With respect to the information Alice and Bob obtain, this EPR version is *identical* to the original protocol $\mathcal{C}^{\alpha}_{\mathtt{pkB}}(\Pi)$: the only difference is the point in time when Alice obtains certain information. Furthermore, we can also do the following modification without affecting (1). Instead of measuring her qubits in T in *her* basis $\theta|_T$, she measures them in *Bob's* basis $\hat{\theta}|_T$; however, she still verifies only whether $x_i = \hat{x}_i$ for those $i \in T$ with $\theta_i = \hat{\theta}_i$. Because the positions $i \in T$ with $\theta_i \neq \hat{\theta}_i$ are not used in the protocol at all, this change has no effect. As the commitment scheme is unconditionally binding if key $\mathtt{pkB}$ is used, Bob's basis $\hat{\theta}$ is well defined by his commitments (although hard to compute), even if Bob is dishonest. The resulting scheme is given in Figure 4.

PROTOCOL EPR-$\mathcal{C}^{\alpha}_{\mathtt{pkB}}(\Pi)$

Preparation: A prepares m EPR pairs and sends the second qubit in each pair to Bob while keeping the others in register $A = A_1 \cdots A_m$. B chooses $\hat{\theta} \in_R \{+, \times\}^m$ and obtains $\hat{x} \in \{0,1\}^m$ by measuring the received qubits in basis $\hat{\theta}$.

Verification:
1. B commits to $\hat{\theta}$ and $\hat{x}$ position-wise: $c_i := \mathtt{Commit}\big((\hat{\theta}_i, \hat{x}_i), r_i\big)$ with randomness r_i for $i = 1, \ldots, m$. He sends the commitments to A.
2. A sends a random test subset $T \subset \{1, \ldots, m\}$ of size αm. B opens c_i for all $i \in T$. A chooses $\theta \in_R \{+, \times\}^m$, measures registers A_i with $i \in T$ in basis $\hat{\theta}_i$ to obtain x_i, and she checks that the openings were correct and that $x_i = \hat{x}_i$ whenever $\theta_i = \hat{\theta}_i$ for $i \in T$. If all tests are passed, A accepts, otherwise, she rejects and aborts the protocol.
3. A measures the remaining registers in basis $\theta|_{\bar{T}}$ to obtain $x|_{\bar{T}}$. The tested positions are discarded by both parties: A and B restrict x and θ, respectively $\hat{\theta}$ and $\hat{x}$, to the positions $i \in \bar{T}$.

Post-processing: As in Π (with $x, \theta, \hat{x}$ and $\hat{\theta}$ restricted to the positions $i \in \bar{T}$).

Fig. 4. EPR version of $\mathcal{C}^{\alpha}_{\mathtt{pkB}}(\Pi)$.

We consider an execution of the scheme from Figure 4 with an honest Alice A and a dishonest Bob B', and we fix $\hat{\theta}$ and $\hat{x}$, determined by Bob's commitments. Let $|\varphi_{AE}\rangle \in \mathcal{H}_A \otimes \mathcal{H}_E$ be the state of the joint system right before Step 2 of the verification phase. Since in the end, we are anyway interested in the pointwise purification of Bob's state, we may indeed assume this state to be pure; if it is not, then we purify it and carry the purifying register R along with E. Clearly, if B' had honestly done his measurements then $|\varphi_{AE}\rangle = |\hat{x}\rangle_{\hat{\theta}} \otimes |\varphi_E\rangle$ for some $|\varphi_E\rangle \in \mathcal{H}_E$. In this case, the quantum memory E would be empty: $H_0(|\varphi_E\rangle\langle\varphi_E|) = 0$. Moreover, X, obtained by measuring $A|_{\bar{T}}$ in basis $\theta|_{\bar{T}}$, would contain $d_H(\theta|_{\bar{T}}, \hat{\theta}|_{\bar{T}})$ random bits. We show that the verification phase enforces these properties, at least approximately in the sense of (1), for an arbitrary dishonest Bob B'.

In the following, $r_H(\cdot, \cdot)$ denotes the relative Hamming distance between two strings, i.e., the Hamming distance divided by their length. Recall that $T \subset \{1, \ldots, m\}$ is random subject to $|T| = \alpha m$. Furthermore, for a fixed $\hat{\theta}$ but a randomly chosen θ, the subset $T' = \{i \in T : \theta_i = \hat{\theta}_i\}$ is a random subset (of arbitrary size) of T. Let the random variable *Test* describe the choice of $test = (T, T')$ as specified above, and consider the state

$$\rho_{TestAE} = \rho_{Test} \otimes |\varphi_{AE}\rangle\langle\varphi_{AE}| = \sum_{test} P_{Test}(test)|test\rangle\langle test| \otimes |\varphi_{AE}\rangle\langle\varphi_{AE}|$$

consisting of the classical *Test* and the quantum state $|\varphi_{AE}\rangle$.

Lemma 4.3. *For any $\varepsilon > 0$, $\hat{x} \in \{0,1\}^m$ and $\hat{\theta} \in \{+, \times\}^m$, the state ρ_{TestAE} is negligibly close (in m) to a state*

$$\tilde{\rho}_{TestAE} = \sum_{test} P_{Test}(test)|test\rangle\langle test| \otimes \big|\tilde{\varphi}^{test}_{AE}\big\rangle\big\langle\tilde{\varphi}^{test}_{AE}\big|$$

where for any $test = (T, T')$*:*

$$\big|\tilde{\varphi}_{AE}^{test}\big\rangle = \sum_{x \in B_{test}} \alpha_x^{test} |x\rangle_{\hat{\theta}} |\psi_E^x\rangle$$

for $B_{test} = \{x \in \{0,1\}^m \,|\, r_H(x|_{\bar{T}}, \hat{x}|_{\bar{T}}) \leq r_H(x|_{T'}, \hat{x}|_{T'}) + \varepsilon\}$ *and arbitrary coefficients* $\alpha_x^{test} \in \mathbb{C}$*.*

In other words, we are close to a situation where for *any* choice of T and T' and for *any* outcome $x|_T$ when measuring $A|_T$ in basis $\hat{\theta}|_T$, the relative error $r_H(x|_{T'}, \hat{x}|_{T'})$ gives an upper bound (which holds with probability 1) on the relative error $r_H(x|_{\bar{T}}, \hat{x}|_{\bar{T}})$ one would obtain by measuring the remaining subsystems A_i with $i \in \bar{T}$ in basis $\hat{\theta}_i$.

Proof. For any *test* we let $|\tilde{\varphi}_{AE}^{test}\rangle$ be the renormalized projection of $|\varphi_{AE}\rangle$ into the subspace $\mathrm{span}\{|x\rangle_{\hat{\theta}} \,|\, x \in B_{test}\} \otimes \mathcal{H}_E$ and let $|\tilde{\varphi}_{AE}^{test\perp}\rangle$ be the renormalized projection of $|\varphi_{AE}\rangle$ into the orthogonal complement, such that $|\varphi_{AE}\rangle = \varepsilon_{test} |\tilde{\varphi}_{AE}^{test}\rangle + \varepsilon_{test}^{\perp} |\tilde{\varphi}_{AE}^{test\perp}\rangle$ with $\varepsilon_{test} = \langle \tilde{\varphi}_{AE}^{test} | \varphi_{AE}\rangle$ and $\varepsilon_{test}^{\perp} = \langle \tilde{\varphi}_{AE}^{test\perp} | \varphi_{AE}\rangle$. By construction, $|\tilde{\varphi}_{AE}^{test}\rangle$ is of the form required in the statement of the lemma. A basic property of the trace norm of pure states gives

$$\delta\big(|\varphi_{AE}\rangle\langle\varphi_{AE}|, |\tilde{\varphi}_{AE}^{test}\rangle\langle\tilde{\varphi}_{AE}^{test}|\big) = \sqrt{1 - |\langle \tilde{\varphi}_{AE}^{test} | \varphi_{AE}\rangle|^2} = |\varepsilon_{test}^{\perp}| \,.$$

This last term corresponds to the square root of the probability, when given *test*, to observe a string $x \notin B_{test}$ when measuring subsystem A of $|\varphi_{AE}\rangle$ in basis $\hat{\theta}$. Furthermore, using elementary properties of the trace norm and Jensen's inequality gives

$$\begin{aligned}
\delta\big(\rho_{TestAE}, \tilde{\rho}_{TestAE}\big)^2 &= \Big(\sum_{test} P_{Test}(test)\, \delta\big(|\varphi_{AE}\rangle\langle\varphi_{AE}|, |\tilde{\varphi}_{AE}^{test}\rangle\langle\tilde{\varphi}_{AE}^{test}|\big) \Big)^2 \\
&= \Big(\sum_{test} P_{Test}(test)\, |\varepsilon_{test}^{\perp}| \Big)^2 \leq \sum_{test} P_{Test}(test)\, |\varepsilon_{test}^{\perp}|^2 \,,
\end{aligned}$$

where the last term is the probability to observe a string $x \notin B_{test}$ when choosing *test* according to P_{Test} and measuring subsystem A of $|\varphi_{AE}\rangle$ in basis $\hat{\theta}$. This situation, though, is a classical sampling problem, for which it is well known that for any measurement outcome x, the probability (over the choice of *test*) that $x \notin B_{test}$ is negligible in m (see e.g. [Hoe63]). □

In combination with Lemma 2.3 on "small superpositions of product states", and writing h for the binary entropy function $h(\mu) = -\big(\mu \log(\mu) + (1-\mu)\log(1-\mu)\big)$ as well as using that $\big|\{y \in \{0,1\}^n \,|\, d_H(y, \hat{y}) \leq \mu n\}\big| \leq 2^{h(\mu)n}$ for any $\hat{y} \in \{0,1\}^n$ and $0 \leq \mu \leq \frac{1}{2}$, we can conclude the following.

Corollary 4.4. *Let* $\tilde{\rho}_{TestAE}$ *be of the form as in Lemma 4.3 (for given* ε*,* $\hat{x}$ *and* $\hat{\theta}$*). For any fixed* $test = (T, T')$ *and for any fixed* $x|_T \in \{0,1\}^{\alpha m}$ *with* $err := r_H(x|_{T'}, \hat{x}|_{T'}) \leq \frac{1}{2}$*, let* $|\psi_{AE}\rangle$ *be the state to which* $|\tilde{\varphi}_{AE}^{test}\rangle$ *collapses when for every* $i \in T$ *subsystem* A_i *is measured in basis* $\hat{\theta}_i$ *and* x_i *is observed, where we understand* A *in* $|\psi_{AE}\rangle$ *to be restricted to the registers* A_i *with* $i \in \bar{T}$*. Finally, let* $\sigma_E = \mathrm{tr}_A(|\psi_{AE}\rangle\langle\psi_{AE}|)$ *and let the random variable* X *describe the outcome when measuring the remaining* $n = (1-\alpha)m$ *subsystems of* A *in basis* $\theta|_{\bar{T}} \in \{+, \times\}^n$*. Then, for any subset* $I \subseteq \{1, \ldots, n\}$ *and any* $x|_I$*,*[12]

$$H_\infty\big(X|_I \,\big|\, X|_{\bar{I}} = x|_{\bar{I}}\big) \geq d_H\big(\theta|_I, \hat{\theta}|_I\big) - h(err + \varepsilon)n \quad \textit{and} \quad H_0\big(\sigma_E\big) \leq h(err + \varepsilon)n \,.$$

Thus, the number of errors between the measured $x|_{T'}$ and the given $\hat{x}|_{T'}$ gives us a bound on the min-entropy of the outcome when measuring the remaining subsystems of A, and on the max-entropy of the state of subsystem E.

[12] Below, $\theta|_I$ (and similarly $\hat{\theta}|_I$) should be understood as first restricting the m-bit vector θ to $\bar{T}$, and then restricting the resulting n-bit vector $\theta|_{\bar{T}}$ to I: $\theta|_I := (\theta|_{\bar{T}})|_I$.

Proof. To simplify notation, we write $\vartheta = \theta|_{\bar{T}}$ and $\hat{\vartheta} = \hat{\theta}|_{\bar{T}}$. By definition of $\tilde{\rho}_{TestAE}$, for any fixed values of $\varepsilon, \hat{x}$, and $\hat{\theta}$, the state $|\psi_{AE}\rangle$ is of the form $|\psi_{AE}\rangle = \sum_{y\in\mathcal{Y}} \alpha_y |y\rangle_{\hat{\vartheta}} \otimes |\psi_E^y\rangle$, where $\mathcal{Y} = \{y \in \{0,1\}^n : d_H(y, \hat{x}|_{\bar{T}}) \leq err + \varepsilon\}$. Consider the corresponding mixture $\tilde{\sigma}_{AE} = \sum_{y\in\mathcal{Y}} |\alpha_y|^2 |y\rangle_{\hat{\vartheta}}\langle y|_{\hat{\vartheta}} \otimes |\psi_E^y\rangle\langle\psi_E^y|$ and define $\tilde{X}$ as the random variable for the outcome when measuring register A of $\tilde{\sigma}_{AE}$ in basis ϑ. Notice that $H_\infty(\tilde{X}) \geq d_H(\vartheta, \hat{\vartheta})$ since any state $|y\rangle_{\hat{\vartheta}}$, when measured in basis ϑ, produces a random bit for every position i with $\vartheta \neq \hat{\vartheta}$. Lemma 2.3 allows us to conclude that $H_\infty(X) \geq H_\infty(\tilde{X}) - \log|\mathcal{Y}| \geq d_H(\vartheta, \hat{\vartheta}) - h(err + \varepsilon)n$ and $H_0(\sigma_E) \leq \log|\mathcal{Y}| \leq h(err + \varepsilon)n$. This proves the claim for $I = \{1, \ldots, n\}$. For arbitrary $I \subset \{1, \ldots, n\}$ and $x|_I$, we can consider the pure state obtained by measuring the registers A_i with $i \notin I$ in basis ϑ_i when $x|_{\bar{I}}$ is observed. This state is still a superposition of at most $|\mathcal{Y}|$ vectors and thus we can apply the exact same reasoning to obtain (1). □

The claim to be shown now follows by combining Lemma 4.3 and Corollary 4.4. Indeed, the ideal state ρ_{Ideal} we promised is produced by putting A and B' in the state $\tilde{\rho}_{TestAE}$ defined in Lemma 4.3, and running Steps 2 and 3 of the verification phase. This state is negligibly close to the real state since by Lemma 4.3 we were negligibly close to the real state before these operations. Corollary 4.4 guarantees that (1) is satisfied.

5 Extensions and Generalizations

5.1 In the Presence of Noise

In the description of the compiler $\mathcal{C}^\alpha$ and in its analysis, we assumed the quantum communication to be noise-free. Indeed, if the quantum communication is noisy honest Alice is likely to reject an execution with honest Bob. It is straightforward to generalize the result to noisy quantum communication: In Step 2 in the verification phase of $\mathcal{C}^\alpha(\Pi)$, Alice rejects and aborts if the relative number of errors between x_i and $\hat{x}_i$ for $i \in T$ with $\theta_i = \hat{\theta}_i$ exceeds the error probability ϕ induced by the noise in the quantum communication by some small $\varepsilon' > 0$. By Hoeffding's inequality [Hoe63], this guarantees that honest Alice does not reject honest Bob except with exponentially small probability. Furthermore, proving the security of this "noise-resistant" compiler goes along the exact same lines as for the original compiler. The only difference is that when applying Corollary 4.4, the parameter err has to be chosen as $err = \phi + \varepsilon'$, so that (1) holds for $\beta = h(err + \varepsilon) = h(\phi + \varepsilon' + \varepsilon)$ and thus the claim of Theorem 4.2 hold for any $\beta > h(\phi)$ (by choosing $\varepsilon, \varepsilon' > 0$ small enough). This allows us to generalize the results from the Section 6 to the setting of noisy quantum communication.

5.2 Bounded-Quantum-Storage Security

In this section we show that our compiler preserves security in the bounded-quantum-storage model (BQSM). In this model, one of the players (Bob in our case) is assumed be able to store only a limited number of qubits beyond a certain point in the protocol. BQSM-secure OT and identification protocols are known [DFR+07,DFSS07], but they can be efficiently broken if the memory bound does not hold. Therefore, by the theorem below, applying the compiler produces protocols with better security, namely the adversary needs large quantum storage *and* large computing power to succeed.

Consider a BB84-type protocol Π, and for a constant $0 < \gamma < 1$, let $\mathfrak{B}^\gamma_{\text{BQSM}}(\Pi)$ be the set of dishonest players B' that store only γn qubits after a certain point in Π, where n is the number of qubits sent initially. Protocol Π is said to be unconditionally secure against γ-BQSM Bob, if it satisfies Definition 3.3 with the restriction that the quantification is over all dishonest $\mathsf{B}' \in \mathfrak{B}^\gamma_{\text{BQSM}}(\Pi)$.

Theorem 5.1. *If Π is unconditionally secure against γ-BQSM Bob, then $\mathcal{C}^\alpha(\Pi)$ (for an $0 < \alpha < 1$) is unconditionally secure against $\gamma(1-\alpha)$-BQSM Bob.*

Proof. Exactly as in the proof of Theorem 4.2, given dishonest Bob B' attacking $\mathcal{C}^\alpha(\Pi)$, we construct dishonest Bob $\mathsf{B}'_\circ$ attacking the original protocol Π. The only difference here is that we let $\mathsf{B}'_\circ$ generate the CRS "correctly" as $\mathtt{pkH}$ sampled according to $\mathcal{G}_{\mathtt{H}}$. It follows by construction of $\mathsf{B}'_\circ$ that $out^{\mathcal{C}^\alpha(\Pi)}_{\mathsf{A},\mathsf{B}'} = out^{\Pi}_{\mathsf{A}_\circ,\mathsf{B}'_\circ}$. Also, it follows by construction of $\mathsf{B}'_\circ$ that if $\mathsf{B}' \in \mathfrak{B}^{\gamma(1-\alpha)}_{\mathrm{BQSM}}(\mathcal{C}^\alpha(\Pi))$ then $\mathsf{B}'_\circ \in \mathfrak{B}^{\gamma}_{\mathrm{BQSM}}(\Pi)$, since $\mathsf{B}'_\circ$ requires the same amount of quantum storage as B' but communicates an α-fraction fewer qubits. It thus follows that there exists $\hat{\mathsf{B}}'$ such that $out^{\Pi}_{\mathsf{A}_\circ,\mathsf{B}'_\circ} \overset{s}{\approx} out^{\mathcal{F}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'}$. This proves the claim. □

5.3 Efficient Simulation

The security definitions we use here are clearly closely related to the UC-security concept in that they require a protocol to implement a certain functionality, and that this can be demonstrated via a simulation argument. However, our definitions do not imply UC-security. For this we would need all simulators to be efficient, and our definition of unconditional security against dishonest Alice does not require this (unlike the definition of computational security against Bob).

Of course, it might still be the case that our compilation preserves efficiency of the simulator, namely if protocol Π is secure against dishonest Alice with efficient simulator $\hat{\mathsf{A}}'$, then so is $\mathcal{C}^\alpha(\Pi)$.

Although this would be desirable, it does not seem to be the case for our basic construction: In order to show such a result, we would need to simulate the preprocessing phase against dishonest A' efficiently and without measuring the qubits that are not "opened during" preprocessing. Once this is done, we can give the remaining qubits to $\hat{\mathsf{A}}'$ who can simulate the rest of the protocol.

However, the whole point of the preprocessing is to ensure that Bob measures all qubits, unless he can break the binding property of the commitments, so the only hope is to bring the simulator in a situation where it can make commitments and open them any way it wants. The standard way to do this is to give the simulator some trapdoor information related to the common reference string, that Bob would not have in real life. Indeed, with such a trapdoor commitment scheme, simulation of the preprocessing is trivial: We just wait until Alice reveals the bases and the test subset, measure qubits in the test subset, and open the commitments according to the measurement results.

While no such trapdoor is known for the commitment scheme we suggested earlier, it is possible to extend the construction efficiently to build in such a trapdoor:

To do this, we need a new ingredient, namely a relation R representing a hard problem, and a Σ-protocol for R. The relation is a set of pairs $R = \{(u, w)\}$ where u can be thought of as a problem instance and w as the solution. The relation is hard if one can efficiently generate $(u, w) \in R$ such that from u one cannot in polynomial time compute w such that $(u, w) \in R$. We need that R is hard even for quantum algorithms. We also need that there is a Σ-protocol, i.e. an honest verifier perfect zero-knowledge interactive proof of knowledge where a prover shows, on input u, that he knows w such that $(u, w) \in R$. Protocol conversations have form (a, b, z) where the prover sends a, the verifier gives a random challenge bit b and the prover sends z. It is required that, given conversations $(a, 0, z_0), (a, 1, z_1)$ that the verifier would accept, one can compute w such that $(u, w) \in R$.

As an example, one can think of $u = (G_0, G_1)$ where G_0, G_1 are isomorphic graphs and w is an isomorphism. The Σ- protocol is just the well-known standard zero-knowledge proof for graph isomorphism. There are several plausible and practically more useful examples, see [DFS04].

Given this, and a commitment scheme with public key $\mathtt{pkH}$ as described above, we build a new commitment scheme as follows: the public key is $u, \mathtt{pkH}$. To commit to a bit b, the committer runs the honest verifier simulator to get a conversation (a, b, z). The commitment is now a, c_0, c_1,

where $c_b = \texttt{Commit}(z, r)$ and $c_{1-b} = \texttt{Commit}(0, r')$. To open a commitment, one reveals b and opens c_b. The receiver checks that (a, b, z) is accepting and that c_b was correctly opened.

By perfect honest verifier zero-knowledge and perfect hiding of commitments based on $\texttt{pkH}$, the new commitment is perfectly hiding. However, if one knows w such that $(u, w) \in R$, one can compute $(a, 0, z_0)$ and $(a, 1, z_1)$ both of which are accepting conversations, and set $c_0 = \texttt{Commit}(z_0, r_0)$, $c_1 = \texttt{Commit}(z_1, r_1)$, and it is now possible to open both ways. Hence w serves as the trapdoor we need for efficient simulation above.

The new commit scheme still has the property we need for the compilation, namely one can choose the public key in a different but indistinguishable way, such that the committed bit can be extracted: we let the public key be $u, \texttt{pkB}$, where $\texttt{pkB}$ is a binding public key for our original scheme. Now, given a commitment (a, c_0, c_1), we can decrypt c_0, c_1 to see which of them contains a valid reply in the Σ-protocol. The only way we can fail to predict how the commitment can be opened is if both c_0 and c_1 contain valid replies. But this would imply that the committer can compute w, so for a polynomial-time bounded committer, this only happens with negligible probability, since the relation is assumed to be hard.

5.4 Doing without a Common Reference String

We can get rid of the CRS assumption by instead generating a reference string from scratch using a coin-flip protocol. In [DL09], such a coin-flip protocol is described and proved secure against quantum adversaries using Watrous' quantum rewinding method [Wat06]. Note that for our compiler, we want the CRS to be an unconditionally hiding public key, and when using Regev's cryptosystem, a uniformly random string (as output by the coin-flip) does indeed determine such a key, except with negligible probability.

6 Applications

6.1 Oblivious Transfer

We discuss a protocol that securely implements one-out-of-two oblivious transfer of strings of length ℓ (i.e. 1-2 OT^ℓ). In 1-2 OT^ℓ, the sender A sends two l-bit strings s_0 and s_1 to the receiver B. B can choose which string to receive (s_k) but does not learn anything about the other one (s_{1-k}). On the other hand, A does not learn B's choice bit k. The protocol is almost identical to the 1-2 OT^1 introduced in [BBCS91], but uses hash functions instead of parity values to mask the inputs s_0 and s_1. The resulting scheme, called $\texttt{1-2 QOT}^\ell$, is presented in Figure 5, where $\mathcal{F}$ denotes a suitable family of universal hash functions with range $\{0,1\}^\ell$ (as specified in [DFR+07]). We assume that $\ell = \lfloor \lambda n \rfloor$ for some constant $\lambda > 0$.

PROTOCOL $\texttt{1-2 QOT}^\ell$:

Preparation: A chooses $x \in_R \{0,1\}^n$ and $\theta \in_R \{+, \times\}^n$ and sends $|x\rangle_\theta$ to B, and B chooses $\hat{\theta} \in_R \{0,1\}^n$ and obtains $\hat{x} \in \{0,1\}^n$ by measuring $|x\rangle_\theta$ in basis $\hat{\theta}$.

Post-processing:
1. A sends θ to B.
2. B partitions all positions $1 \leq i \leq n$ in two subsets according to his choice bit $k \in \{0,1\}$: the "good" subset $I_k := \{i : \theta_i = \hat{\theta}_i\}$ and the "bad" subset $I_{1-k} := \{i : \theta_i \neq \hat{\theta}_i\}$. B sends (I_0, I_1) to A.
3. A sends descriptions of $f_0, f_1 \in_R \mathcal{F}$ together with $m_0 := s_0 \oplus f_0(x|_{I_0})$ and $m_1 := s_1 \oplus f_1(x|_{I_1})$.
4. B computes $s_k = m_k \oplus f_k(\hat{x}|_{I_k})$.

Fig. 5. Protocol for String OT.

Theorem 6.1. *Protocol* $\texttt{1-2}\,\texttt{QOT}^\ell$ *is unconditionally secure against* β*-benign Bob for any* $\beta < \frac{1}{8} - \frac{\lambda}{2}$.

Proof. Let $\mathcal{F}_{OT^\ell}$ be the ideal oblivious transfer functionality. For any given benign Bob B', we construct $\hat{\mathsf{B}}'$ the following way. $\hat{\mathsf{B}}'$ runs locally a copy of B' and simulates Alice by running A up to but not including Step 3. After the preparation phase, $\hat{\mathsf{B}}'$ gets $\hat{\theta}$ since B' is benign. When the simulation of A reaches the point just after the announcement of f_0 and f_1 in Step 3, $\hat{\mathsf{B}}'$ finds k' such that $d_H(\hat{\theta}|_{I_{k'}}, \theta|_{I_{k'}})$ is minimum for $k' \in \{0,1\}$. $\hat{\mathsf{B}}'$ then calls $\mathcal{F}_{OT^\ell}$ with input k' and obtains output $s_{k'}$. $\hat{\mathsf{B}}'$ sets $m'_{k'} = s_{k'} \oplus f_{k'}(x|_{I_{k'}})$ and $m'_{1-k'} \in_R \{0,1\}^\ell$ before sending (m_0, m_1) to B'. $\hat{\mathsf{B}}'$ then outputs whatever B' outputs.

We now argue that the state output by $\hat{\mathsf{B}}'$ is statistically close to the state output by B' when executing $\texttt{1-2}\,\texttt{QOT}^\ell$ with the real A. The only difference is that while $\hat{\mathsf{B}}'$ outputs $m'_{1-k'} \in_R \{0,1\}^\ell$, B' outputs $m_{1-k'} = s_{1-k'} \oplus f_{1-k'}(x|_{I_{1-k'}})$. To conclude, we simply need to show that $m_{1-k'}$ is statistically indistinguishable from uniform from the point of view of B'. Note that since θ and $\hat{\theta}$ are independent and θ is a uniform n-bit string, we have that for any $\epsilon > 0$, $d_H(\theta, \hat{\theta}) > (1-\epsilon)n/2$, except with negligible probability. It follows that with overwhelming probability $d_H(\theta|_{I_{1-k'}}, \hat{\theta}|_{I_{1-k'}}) \geq (1-\epsilon)n/4$. Since B' is β-benign, we have that $H_\infty\big(X|_{I_{1-k'}} \,\big|\, X|_{I_{k'}} = x|_{I_{k'}}\big) \geq (1-\epsilon)n/4 - \beta n$ and $H_0(\rho_E) \leq \beta n$ which implies, from privacy amplification, that $f_{1-k'}(x|_{I_{1-k'}})$ is statistically indistinguishable from uniform for B' provided $\frac{\ell}{n} < \frac{1}{4} - 2\beta - \epsilon$ for any $\epsilon > 0$. We conclude that $m_{1-k'}$ is statistically close to uniform. □

By combining Theorem 6.1 with Theorem 4.2, and the results of [DFR+07] (realizing that the same analysis also applies to $\texttt{1-2}\,\texttt{QOT}^\ell$) with Theorem 5.1, we obtain the following hybrid-security result.

Corollary 6.2. *Let* $0 < \alpha < 1$ *and* $\lambda < \frac{1}{8}$. *Then protocol* $\mathcal{C}^\alpha(\texttt{1-2}\,\texttt{QOT}^\ell)$ *is computationally secure against dishonest Bob and unconditionally secure against* $\gamma(1-\alpha)$*-BQSM Bob with* $\gamma < \frac{1}{4} - 2\lambda$.

6.2 Password-Based Identification

We want to apply our compiler to the quantum password-based identification scheme from [DFSS07]. Such an identification scheme allows a user A to identify herself to server B by means of a common (possibly non-uniform and low-entropy) password $w \in \mathcal{W}$, such that dishonest A' cannot delude honest server B with probability better then trying to guess the password, and dishonest B' learns no information on A's password beyond trying to guessing it and learn whether the guess is correct or not.

In [DFSS07], using quantum-information-theoretic security definitions, the proposed identification scheme was proven to be unconditionally secure against arbitrary dishonest Alice and against quantum-memory-bounded dishonest Bob. In [FS09] it was then shown that these security definitions imply simulation-based security as considered here, with respect to the functionality $\mathcal{F}_{ID}$ given in Figure 6.[13]

Functionality $\mathcal{F}_{ID}$: Upon receiving $w_A, w_B \in \mathcal{W}$ from user Alice and from server Bob, respectively, $\mathcal{F}_{ID}$ outputs the bit $y := (w_A \stackrel{?}{=} w_B)$ to Bob. In case Alice is dishonest, she may choose $w_A = \perp$ (where $\perp \notin \mathcal{W}$). For any choice of w_A the bit y is also output to dishonest Alice.

Fig. 6. The Ideal Password-Based Identification Functionality.

[13] Actually, the definition and proof from [DFSS07] guarantees security only for a slightly weaker functionality, which gives some unfair advantage to dishonest A' in case she guesses the password correctly; however, as discussed in [FS09], the protocol from [DFSS07] does implement functionality $\mathcal{F}_{ID}$.

We cannot directly apply our compiler to the identification scheme as given in [DFSS07], since it is *not* a BB84-type protocol. The protocol does start with a preparation phase in which Alice sends BB84 qubits to Bob, but Bob does not measure them in a random basis but in a basis determined by his password $w_B \in \mathcal{W}$; specifically, Bob uses as basis the encoding $\mathfrak{c}(w_B)$ of w_B with respect to a code $\mathfrak{c} : \mathcal{W} \to \{+,\times\}^n$ with "large" minimal distance. However, it is easy to transform the original protocol from [DFSS07] into a BB84-type protocol without affecting security: We simply let Bob apply a *random shift* κ to the code, which Bob only announces to Alice in the post-processing phase, and then Alice and Bob complete the protocol with the shifted code. The resulting protocol QID is described in Figure 7, where $\mathcal{F}$ and $\mathcal{G}$ are suitable families of (strongly) universal hash functions (we refer to [DFSS07] for the exact specifications). It is not hard to see that this modification does not affect security as proven in [DFSS07] (and [FS09]).

PROTOCOL QID :

Preparation: A chooses $x \in_R \{0,1\}^n$ and $\theta \in_R \{+,\times\}^n$ and sends $|x\rangle_\theta$ to B, and B chooses $\hat{\theta} \in_R \{0,1\}^n$ and obtains $\hat{x} \in \{0,1\}^n$ by measuring $|x\rangle_\theta$ in basis $\hat{\theta}$.

Post-processing:
1. B computes a string $\kappa \in \{+,\times\}^n$ such that $\hat{\theta} = \mathfrak{c}(w) \oplus \kappa$ (we think of $+$ as 0 and $\times$ as 1 so that $\oplus$ makes sense). He sends κ to A and we define $\mathfrak{c}'(w) := \mathfrak{c}(w) \oplus \kappa$.
2. A sends θ and $f \in_R \mathcal{F}$ to B. Both compute $I_w := \{i : \theta_i = \mathfrak{c}'(w)_i\}$.
3. B sends $g \in_R \mathcal{G}$.
4. A sends $z := f(x|_{I_w}) \oplus g(w)$ to B.
5. B accepts if and only if $z = f(\hat{x}|_{I_w}) \oplus g(w)$.

Fig. 7. Protocol for Secure Password-Based Identification

Theorem 6.3. *If the code $\mathfrak{c} : \mathcal{W} \to \{+,\times\}^n$ can correct at least δn errors in polynomial-time for a constant δ, then protocol* QID *is unconditionally secure against β-benign Bob for any $\beta < \frac{\delta}{4}$.*

Proof. For any given benign Bob B', we construct $\hat{\mathsf{B}}'$ as follows. $\hat{\mathsf{B}}'$ runs locally a copy of B' and simulates Alice's actions by running A faithfully except for the following modifications. After the preparation phase, $\hat{\mathsf{B}}'$ gets $\hat{\theta}$ and κ from B' and attempts to decode $\hat{\theta} \oplus \kappa$. If this succeeds, it computes w' such that $\mathfrak{c}(w')$ is the decoded codeword. Otherwise an arbitrary w' is chosen. Then, $\hat{\mathsf{B}}'$ submits w' as Bob's input w_B to $\mathcal{F}_{ID}$ and receives output $y \in \{0,1\}$. If $y = 1$ then $\hat{\mathsf{B}}'$ faithfully completes A's simulation using w' as w; else, $\hat{\mathsf{B}}'$ completes the simulation by using a random z' instead of z. In the end, $\hat{\mathsf{B}}'$ outputs whatever B' outputs.

We need to show that the state output by $\hat{\mathsf{B}}'$ (respectively B') above is statistically close to the state output by B' when executing QID with real A. Note that if $w' = w_A$, then the simulation of A is perfect and thus the two states are equal. If $w' \neq w_A$ then the simulation is not perfect: the real A would use $z = f(x|_{I_{w_A}}) \oplus g(w_A)$ instead of random z'. It thus suffices to argue that $f(x|_{I_w})$ is statistically close to random and independent of the view of B' for any fixed $w \neq w'$. Note that this is also what had to be proven in [DFSS07], but under a different assumption, namely that B' has bounded quantum memory, rather than that he is benign; nevertheless, we can recycle part of the proof.

Recall from the definition of a benign Bob that the common state after the preparation phase is statistically close to a state for which it is guaranteed that $H_\infty(X|_I) \geq d_H(\theta|_I, \hat{\theta}|_I) - \beta n$ for any $I \subseteq \{1,\ldots,n\}$, and $H_0(\rho_{ER}) \leq \beta n$. By the closeness of these two states, switching from the real state to the "ideal" state (which satisfies these bounds) has only a negligible effect on the state output by $\hat{\mathsf{B}}'$; thus, we may assume these bounds to hold.

Now, if decoding of $\hat{\theta} \oplus \kappa$ succeeded, it is at Hamming distance at most δn from $\mathfrak{c}(w')$. Since the distance from here to the (distinct) codeword $\mathfrak{c}(w)$ is greater than $2\delta n$, we see that

$\hat{\theta} \oplus \kappa$ is at least δn away from $\mathfrak{c}(w)$. The same is true if decoding failed, since then $\hat{\theta} \oplus \kappa$ is at least δn away from any codeword. It follows that $\mathfrak{c}'(w) = \mathfrak{c}(w) \oplus \kappa$ has Hamming distance at least δn from $\hat{\theta}$. Furthermore, for arbitrary $\varepsilon > 0$ and except with negligible probability, the Hamming distance between $\theta|_{I_w} = \mathfrak{c}'(w)|_{I_w}$ and $\hat{\theta}|_{I_w}$ is at least essentially $(\delta/2 - \varepsilon)n$. Therefore, we can conclude that $H_\infty(X|_{I_w}) \geq (\delta/2 - \varepsilon - \beta)n$ and $H_0(\rho_{ER}) \leq \beta n$. But now, if such bounds hold such that $H_\infty(X|_{I_w}) - H_0(\rho_{ER})$ is positive and linear in n, which is the case here by the choice of parameters, then we can step into the proof from [DFSS07] and conclude by privacy amplification [RK05] that z is close to random and independent of E. This finishes the proof. □

By combining Theorem 6.3 with Theorem 4.2, and the results of [DFSS07] with Theorem 5.1, we obtain the following hybrid-security result.

Corollary 6.4. *Let $0 < \alpha < 1$ and $|\mathcal{W}| \leq 2^{\nu n}$. If the code $\mathfrak{c} : \mathcal{W} \to \{+, \times\}^n$ can correct δn errors for a constant $\delta > 0$ in polynomial-time, then protocol $\mathcal{C}^\alpha(\mathtt{QID})$ is computationally secure against dishonest Bob and unconditionally secure against $\gamma(1-\alpha)$-BQSM Bob with $\gamma < \frac{\delta}{2} - \nu$.*

Families of codes as required in these results, correcting a constant fraction of errors efficiently and with constant information rate are indeed known, see [SS96].

In the next section, we briefly discuss how to obtain hybrid security against *man-in-the-middle* attacks by means of incorporating the techniques used in [DFSS07] to obtain security in the BQSM against such attacks.

6.3 Protecting against Man-in-the-middle Attacks

The compiled quantum protocols from Sections 6.1 and 6.2 protect against (arbitrary) dishonest Alice and against (computationally or quantum-storage bounded) dishonest Bob. However, in particular in the context of identification, it is also important to protect against a *man-in-the-middle* attacker, Eve, who attacks an execution of the protocol with honest parties A and B while having full control over the classical and the quantum communication. Both, $\mathtt{QID}$ and $\mathcal{C}^\alpha(\mathtt{QID})$, are insecure in this model: Eve might measure one of the transmitted qubits, say, in the + -basis, and this way learn information on the basis $\hat{\theta}_i$ used by B and thus on the password w simply by observing if B accepts or rejects in the end.

In [DFSS07] it was shown how to enhance $\mathtt{QID}$ in order to obtain security (in the bounded-quantum-storage model) against man-in-the-middle attacks. The very same techniques can also be used to obtain *hybrid security* against man-in-the-middle attacks for $\mathcal{C}^\alpha(\mathtt{QID})$. The techniques from [DFSS07] consist of the following two add-on's to the original protocol. (1) Checking of a random subset of the qubits in order to detect disturbance of the quantum communication; note that $\mathcal{C}^\alpha(\mathtt{QID})$ already does such a check, so this is already taken care of here. And (2) authentication of the classical communication. This requires that Alice and Bob, in addition to the password, share a high-entropy key k that could be stored, e.g., on a smart-card. This key will be used for a so-called *extractor MAC* which has the additional property, besides being a MAC, that it also acts as an extractor, meaning if the message to be authenticated has high enough min-entropy, then the key-tag pair is close to randomly and independently distributed. As a consequence, the tag gives away (nearly) no information on k and thus k can be re-used in the next execution of the protocol.[14]

Concretely, in order to obtain hybrid-security against man-in-the-middle attacks for $\mathcal{C}^\alpha(\mathtt{QID})$, A will, in her last move of the protocol, use an extractor MAC to compute and send to B an authentication tag, computed on all the classical messages exchanged plus the string $x|_{I_w}$. This tag, together with the qubit checks, prevents Eve from interfering with the (classical and quantum) communication without being detected, and security against Eve essentially

[14] This is in contrast to the standard way of authenticating the classical communication, where the authentication key can only be used a bounded number of times.

follows from the security against impersonation attacks. Note that including the $x|_{I_w}$ into the authenticated message guarantees the necessary min-entropy, and as such the re-usability of the key k.

We emphasize that the protocol is still secure against impersonation attacks (i.e. dishonest Alice or Bob) even if the adversary knows k. We omit formal proofs since they literally follow the corresponding proofs in [DFSS07].

Acknowledgments

We thank Dominique Unruh for useful comments about the efficiency of the simulators.

SF is supported by the Dutch Organization for Scientific Research (NWO). CL acknowledges financial support by the MOBISEQ research project funded by NABIIT, Denmark. LS is supported by the QUSEP project of the Danish Natural Science Research Council, and by the QuantumWorks Network. CS acknowledges support by EU fifth framework project QAP IST 015848 and a NWO VICI grant 2004-2009.

References

[BBCS91] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska. Practical quantum oblivious transfer. In *Advances in Cryptology—CRYPTO '91*, volume 576 of *Lecture Notes in Computer Science*, pages 351–366. Springer, 1991.

[CDMS04] Claude Crépeau, Paul Dumais, Dominic Mayers, and Louis Salvail. Computational collapse of quantum state with application to oblivious transfer. In *Theory of Cryptography Conference (TCC)*, volume 2951 of *Lecture Notes in Computer Science*. Springer, 2004.

[DFR+07] Ivan B. Damgård, Serge Fehr, Renato Renner, Louis Salvail, and Christian Schaffner. A tight high-order entropic quantum uncertainty relation with applications. In *Advances in Cryptology—CRYPTO '07*, volume 4622 of *Lecture Notes in Computer Science*, pages 360–378. Springer, 2007.

[DFS04] Ivan B. Damgård, Serge Fehr, and Louis Salvail. Zero-knowledge proofs and string commitments withstanding quantum attacks. In *Advances in Cryptology—CRYPTO '04*, volume 3152 of *Lecture Notes in Computer Science*, pages 254–272. Springer, 2004.

[DFSS05] Ivan B. Damgård, Serge Fehr, Louis Salvail, and Christian Schaffner. Cryptography in the bounded quantum-storage model. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 449–458, 2005. Full version available at: `http://arxiv.org/abs/quant-ph/0508222v2`.

[DFSS07] Ivan B. Damgård, Serge Fehr, Louis Salvail, and Christian Schaffner. Secure identification and QKD in the bounded-quantum-storage model. In *Advances in Cryptology—CRYPTO '07*, volume 4622 of *Lecture Notes in Computer Science*, pages 342–359. Springer, 2007.

[DFSS08] Ivan B. Damgård, Serge Fehr, Louis Salvail, and Christian Schaffner. Cryptography in the bounded-quantum-storage model. *SIAM Journal on Computing*, 37(6):1865–1890, 2008.

[DL09] Ivan B. Damgård and Carolin Lunemann. Quantum-secure coin-flipping and applications. To appear in *Advances in Cryptology—ASIACRYPT '09*, `http://arxiv.org/abs/0903.3118`, 2009.

[FS09] Serge Fehr and Christian Schaffner. Composing quantum protocols in a classical environment. In *Theory of Cryptography Conference (TCC)*, volume 5444 of *Lecture Notes in Computer Science*, pages 350–367. Springer, 2009.

[Hoe63] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, March 1963.

[KN08] Gillat Kol and Moni Naor. Games for exchanging information. In *Theory of Cryptography Conference (TCC)*, volume 4948 of *Lecture Notes in Computer Science*, pages 423–432. Springer, 2008.

[Lo97] Hoi-Kwong Lo. Insecurity of quantum secure computations. *Physical Review A*, 56(2):1154–1162, 1997.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge university press, 2000.

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In *Advances in Cryptology—CRYPTO '08*, volume 5157 of *Lecture Notes in Computer Science*, pages 554–571. Springer, 2008.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *37th Annual ACM Symposium on Theory of Computing (STOC)*, pages 84–93, 2005.

[Ren05] Renato Renner. *Security of Quantum Key Distribution*. PhD thesis, ETH Zürich (Switzerland), September 2005. `http://arxiv.org/abs/quant-ph/0512258`.

[RK05] Renato Renner and Robert König. Universally composable privacy amplification against quantum adversaries. In *Theory of Cryptography Conference (TCC)*, volume 3378 of *Lecture Notes in Computer Science*, pages 407–425. Springer, 2005.

[SS96] Michael Sipser and Daniel A. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.

[Wat06] John Watrous. Zero-knowledge against quantum attacks. In *38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 296–305, 2006. full version available at `http://www.cs.uwaterloo.ca/~watrous/papers.html`.

[WST08] Stephanie Wehner, Christian Schaffner, and Barbara M. Terhal. Cryptography from noisy storage. *Physical Review Letters*, 100(22):220502, 2008.

[Yao95] Andrew Chi-Chih Yao. Security of quantum protocols against coherent measurements. In *27th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 67–75, 1995.

A Sequential Composition Theorem for Computational Security

In this appendix, we show that our new Definition 3.4 of computational security allows for sequential composability in a classical environment. In order to state the composition theorem, we need to define what we mean by running a quantum protocol in a classical environment. Again, we give here a brief summary of the setting from [FS09] and refer the interested reader to the original article for further details.

A classical two-party *oracle protocol*[15] $\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell} = (\hat{\mathsf{A}}, \hat{\mathsf{B}})$ between Alice and Bob is a protocol which makes a bounded number k of sequential oracle calls to possibly different ideal functionalities $\mathcal{F}_1, \ldots, \mathcal{F}_\ell$.

For the oracle protocol to be *classical*, we mean that it has classical in- and output (for the honest players), but also that all communication between Alice and Bob is classical.[16] Consider a dishonest player, say Bob, and consider the common state $\rho_{U_j V_j'}$ at any point during the execution of the oracle protocol when a call to functionality $\mathcal{F}_i$ is made. The requirement for the oracle protocol to be *classical* is now expressed in that there exists a classical Z_j—to be understood as consisting of $\hat{\mathsf{B}}'$'s classical communication with $\hat{\mathsf{A}}$ and with the $\mathcal{F}_{i'}$'s up to this point—such that given Z_j, Bob's quantum state V_j' is not entangled with Alice's classical input and auxiliary information: $\rho_{U_j Z_j V_j'} = \rho_{U_j \leftrightarrow Z_j \leftrightarrow V_j'}$. Furthermore, we require that we may assume Z_j to be part of V_j' in the sense that for any $\hat{\mathsf{B}}'$ there exists $\hat{\mathsf{B}}''$ such that Z_j is part of V_j'. This definition is motivated by the observation that if Bob can communicate only classically with Alice, then he can entangle his quantum state with information on Alice's side only by means of the classical communication.

We also consider the protocol we obtain by replacing the ideal functionalities by quantum two-party sub-protocols $\pi_1, \ldots, \pi_\ell$ with classical in- and outputs for the honest parties: whenever $\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell}$ instructs $\hat{\mathsf{A}}$ and $\hat{\mathsf{B}}$ to execute $\mathcal{F}_{i\hat{\mathsf{A}},\hat{\mathsf{B}}}$, they instead execute π_i and take the resulting outputs. We write $\Sigma^{\pi_1 \cdots \pi_\ell} = (\mathsf{A}, \mathsf{B})$ for the real quantum protocol we obtain this way.

We recall that in order to define computational security against a dishonest Bob in the common-reference-string model, we considered a polynomial-size quantum circuit, called *input sampler*, which takes as input the security parameter m and the CRS ω (chosen according to its distribution) and which produces the input state $\rho_{UZV'}$; U is Alice's classical input to the protocol, and Z and V' denote the respective classical and quantum information given to dishonest Bob. We require from the input sampler that $\rho_{UZV'} = \rho_{U \leftrightarrow Z \leftrightarrow V'}$, i.e., that V' is correlated with Alice's part only via the classical Z. When considering classical hybrid protocols $\Sigma^{\pi_1 \cdots \pi_\ell}$ in the real world, where the oracle calls are replaced with quantum protocols using a common reference string, it is important that every real protocol π_i uses a separate instance (or part) of the common reference string which we denote by ω_i.

[15] In [FS09], the more standard term *hybrid protocol* is used, but as this term is used differently in this paper, we avoid it here in the context of composability.

[16] We do not explicitly require the internal computations of the honest parties to be classical.

Theorem A.1 (Sequential Composition). *Let $\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell} = (\hat{\mathsf{A}}, \hat{\mathsf{B}})$ be a classical two-party oracle protocol which makes at most $k = poly(n)$ oracle calls to the functionalities, and for every $i \in \{1, \ldots, \ell\}$, let protocol π_i be a computationally secure implementation of $\mathcal{F}_i$ against $\mathfrak{B}_{\text{poly}}$.*

Then, for every real-world adversary $\mathsf{B}' \in \mathfrak{B}_{\text{poly}}$ who accesses the common reference string $\omega = \omega_1, \ldots, \omega_k$ there exists an ideal-world adversary $\hat{\mathsf{B}}' \in \mathfrak{B}_{\text{poly}}$ who does not use ω such that for every efficient input sampler, it holds that the outputs in the real and ideal world are q-indistinguishable, i.e.

$$out^{\Sigma^{\pi_1 \cdots \pi_\ell}}_{\mathsf{A},\mathsf{B}'} \overset{q}{\approx} out^{\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'}$$

Note that we do not specify what it means for the oracle protocol to be secure; Theorem A.1 guarantees that *whatever* the oracle protocol achieves, an indistinguishable output is produced by the real-life protocol with the oracle calls replaced by protocols. But of course in particular, if the oracle protocol *is* secure in the sense of Definition 3.4, then so is the real-life protocol:

Corollary A.2. *If $\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell}$ is a computationally secure implementation of $\mathcal{G}$ against $\mathfrak{B}_{\text{poly}}$, and if π_i is a computationally secure implementation of $\mathcal{F}_i$ against $\mathfrak{B}_{\text{poly}}$ for every $i \in \{1, \ldots, \ell\}$, then $\Sigma^{\pi_1 \cdots \pi_\ell}$ with at most $k = poly(n)$ oracle calls is a computationally secure implementation of $\mathcal{G}$ against $\mathfrak{B}_{\text{poly}}$.*

The following proof is an adaptation of the sequential-composability proof in the information-theoretical setting given in [FS09].

Proof (of Theorem A.1). Consider a dishonest $\mathsf{B}' \in \mathfrak{B}_{\text{poly}}$. We prove the claim by induction on k. If no oracle calls are made, we can set $\hat{\mathsf{B}}' := \mathsf{B}'$ and the claim holds trivially. Consider now a protocol $\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell}$ with at most $k > 0$ oracle calls. For simplicity, we assume that the number of oracle calls equals k, otherwise we instruct the players to makes some "dummy calls". Let $\rho_{U_k Z_k V'_k}$ be the common state right before the k-th and thus last call to one of the sub-protocols $\pi_1, \ldots, \pi_\ell$ in the execution of the real protocol $\Sigma^{\pi_1, \ldots, \pi_\ell}$. To simplify notation in the rest of the proof, we omit the index k and write $\rho_{\bar{U}\bar{Z}\bar{V}'}$ instead; see Figure 8. We know from the induction hypothesis for $k-1$ that there exists an ideal-world adversary $\hat{\mathsf{B}}' \in \mathfrak{B}_{\text{poly}}$ not using the common reference string such that $\rho_{\bar{U}\bar{Z}\bar{V}'} \overset{q}{\approx} \sigma_{\bar{U}\bar{Z}\bar{V}'}$ where $\sigma_{\bar{U}\bar{Z}\bar{V}'}$ is the common state right before the k-th call to a functionality in the execution of the oracle protocol $\Sigma^{\mathcal{F}_1 \cdots \mathcal{F}_\ell}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'} \rho_{UZV'}$. As described at the begin of this section, $\bar{U}$ and $\bar{Z}, \bar{V}'$ are to be understood as follows. $\bar{U}$ denotes A's (respectively $\hat{\mathsf{A}}$'s) input to the sub-protocol (respectively functionality) that is to be called next. $\bar{Z}$ collects the classical communication dictated by $\Sigma^{\mathcal{F}_1, \ldots, \mathcal{F}_\ell}$ as well as $\hat{\mathsf{B}}'$'s classical inputs to and outputs from the previous oracle calls and $\bar{V}'$ denotes the dishonest player's current quantum state. Note that the existence of $\bar{Z}$ is guaranteed by our formalization of *classical* oracle protocols and $\sigma_{\bar{U}\bar{Z}\bar{V}'} = \sigma_{\bar{U} \leftrightarrow \bar{Z} \leftrightarrow \bar{V}'}$.

Let ω_i be the common reference string used in protocol π_i. For simplicity, we assume that the index i, which determines the sub-protocol π_i (functionality $\mathcal{F}_i$) to be called next, is *fixed* and we just write π and $\mathcal{F}$ for π_i and $\mathcal{F}_i$, respectively.

It follows from Definition 3.4 of computational security that there exists $\hat{\mathsf{B}}' \in \mathfrak{B}_{\text{poly}}$ (independent of the input state) not using ω_i such that the corresponding output states $\sigma_{\bar{X}\bar{Z}\bar{Y}'}$ and $\tau_{\bar{X}\bar{Z}\bar{Y}'}$ produced by $\mathcal{F}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'}$ (as prescribed by the oracle protocol) and $\pi_{\mathsf{A},\mathsf{B}'}$ run on the state $\sigma_{\bar{U}\bar{Z}\bar{V}'} = \sigma_{\bar{U} \leftrightarrow \bar{Z} \leftrightarrow \bar{V}'}$ are q-indistinguishable.

The induction step is then completed as follows.

$$out^{\Sigma^{\pi}}_{\mathsf{A},\mathsf{B}'} = \rho_{\bar{X}\bar{Z}\bar{Y}'} = (\pi_{\mathsf{A},\mathsf{B}'})\, \rho_{\bar{U}\bar{Z}\bar{V}'} \overset{q}{\approx} (\pi_{\mathsf{A},\mathsf{B}'})\, \sigma_{\bar{U}\bar{Z}\bar{V}'} = \sigma_{\bar{X}\bar{Z}\bar{Y}'} \overset{q}{\approx} \tau_{\bar{X}\bar{Z}\bar{Y}'} = out^{\Sigma^{\mathcal{F}}}_{\hat{\mathsf{A}},\hat{\mathsf{B}}'}$$

Note that the strategy of $\hat{\mathsf{B}}'$ does not depend on the state $\sigma_{\bar{U}\bar{Z}\bar{V}'}$, and hence, the overall ideal-world adversary $\hat{\mathsf{B}}'$ does not depend on the input state either. Furthermore, the concatenation of two polynomially bounded players is polynomially bounded, i.e. $\hat{\mathsf{B}}' \in \mathfrak{B}_{\text{poly}}$.

□

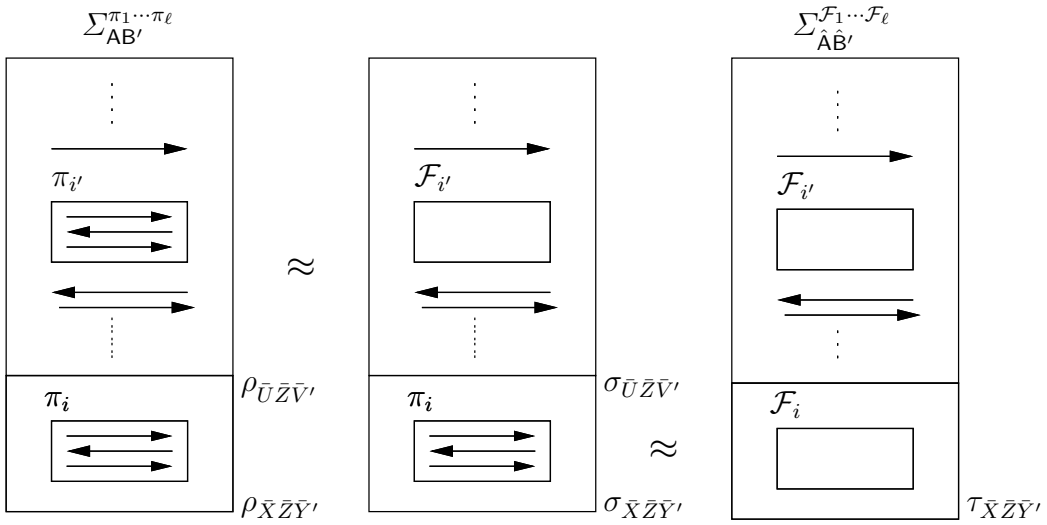

Fig. 8. Steps of the Composability Proof